05/25/00 JC808 U.S. PTO

NEW, CONTINUATION, DIVISIONAL OR CONTINUATION-IN-PART APPLICATION UNDER 37 C.F.R. §1.53(b)

JC714 U.S. PTO 09/578739 05/25/00

Attorney Docket No. 9432-000057

Express Mail Label No. EL 486 600 570 US

Date May 25, 2000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Hon. Commissioner of Patents and Trademarks
Washington, D. C. 20231

Sir:

Transmitted herewith for filing under 37 C.F.R §1.53(b) is a patent application for

PROVIDING VCR FUNCTIONALITY FOR DATA-CENTERED VIDEO MULTICAST

identified by: [ ] First named inventor ______________
or [X] Attorney Docket No. (see above)

1. **Type of Application**

[ ] This application is a new (non-continuing) application.

[X] This application is a [X] continuation / [ ] divisional / [ ] continuation-in-part of prior application **PCT International Serial No. 99/25422**. Amend the specification by inserting before the first line the sentence:

--This is a continuation of patent application **PCT International Serial No. 99/25422**, filed **10/28/99**.--

[X] The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied, is considered part of the disclosure of the accompanying application and is hereby incorporated by reference therein.

If for some reason applicant has not requested a sufficient extension of time in the parent application, and/or has not paid a sufficient fee for any necessary response in the parent application and/or for the extension of time necessary to prevent the abandonment of the parent application prior to the filing of this application, please consider this as a Request for an Extension for the required time period and/or authorization to charge our Deposit Account No. 08-0750 for any fee that may be due. THIS FORM IS BEING FILED IN TRIPLICATE: one copy for this application; one copy for use in connection with the Deposit Account (if applicable); and one copy for the above-mentioned parent application (if any extension of time is necessary).

2. **Contents of Application**

a. Specification of 48 pages;
   [ ] A microfiche computer program (Appendix);
   [ ] A nucleotide and/or amino acid sequence submission;

   [ ] Because the enclosed application is in a non-English language, a verified English translation [ ] is enclosed [ ] will be filed.

   [ ] Cancel original claims _____ of the prior application before calculating the filing fee. (At least one original independent claim must be retained for filing date purposes.)

b. [X] Drawings on 5 sheets;

Attorney Docket No. 9432-000057

Express Mail Label No. EL 486 600 570 US

Date May 25, 2000

c. [X] A signed Oath/Declaration [ ] is enclosed / [X] will be filed in accordance with 37 C.F.R. §1.53(f).

The enclosed Oath/Declaration is [ ] newly executed / [ ] a copy from a prior application under 37 C.F.R. §1.63(d) / [ ] accompanied by a statement requesting the deletion of person(s) not inventors in the continuing application.

d. **Fees**

FILING FEE CALCULATION	Number Filed				Number Extra		Rate		Basic Fee $690.00
Total Claims	31	-	20	=	11	×	$18.00	=	198.00
Independent Claims	5	-	3	=	2	×	$78.00	=	156.00
Multiple Dependent Claim(s) Used							$260.00	=	
FILING FEE - NON-SMALL ENTITY									$1,044.00
FILING FEE - SMALL ENTITY: Reduction by 1/2 .... [ ] Verified Statement under 37 C.F.R. §1.27 is enclosed. [X] Verified Statement filed in prior application.									
Assignment Recordal Fee ($40.00)									
37 C.F.R. §1.17(k) Fee (non-English application)........									
TOTAL									**$1,044.00**

[X] A check is enclosed to cover the calculated fees. The Commissioner is hereby authorized to charge any additional fees that may be required, or credit any overpayment, to Deposit Account No. 08-0750. A duplicate copy of this document is enclosed.

[ ] The calculated fees will be paid within the time allotted for completion of the filing requirements.

[ ] The calculated fees are to be charged to Deposit Account No. 08-0750. The Commissioner is hereby authorized to charge any additional fees that may be required, or credit any overpayment, to said Deposit Account. A duplicate copy of this document is enclosed.

3. **Priority Information**

[ ] **Foreign Priority**: Priority based on ______________________________, is claimed.

[ ] A copy of the above referenced priority document [X] is enclosed / [ ] will be filed in due course, pursuant to 35 U.S.C. §119(a)-(d).

[X] **Provisional Application Priority**: Priority based on United States Provisional Application No. **60/106,167**, filed **10/29/98,** is claimed under 35 U.S.C. §120.

Attorney Docket No. 9432-000057

Express Mail Label No. EL 486 600 570 US

Date May 25, 2000

4. **Other Submissions**

[ ] A Preliminary Amendment is enclosed.

[ ] An Information Disclosure Statement, ____ sheets of PTO Form 1449, and ____ patent(s)/publications/documents are enclosed.

[X] A power of attorney

[ ] is submitted [ ] with the new Oath/Declaration.

[X] is of record in the prior application and [ ] is in the original papers / [X] a copy is enclosed.

[ ] An Assignment of the invention

[ ] is enclosed with a cover sheet pursuant to 37 C.F.R. §§3.11, 3.28 and 3.31.

[ ] is of record in a prior application. The assignment is to ____, and is recorded at Reel ____, Frame(s) ____.

[ ] An Establishment of Assignee's Right To Prosecute Application Under 37 C.F.R. §3.73(b), and Power Of Attorney is enclosed.

[X] An Express Mailing Certificate is enclosed.

[X] Other: acknowledgment postcard.

Attention is directed to the fact that the correspondence address for this application is:

Harness, Dickey & Pierce, P.L.C.
P.O. Box 828
Bloomfield Hills, Michigan 48303
(248) 641-1600.

Respectfully,

Date May 25, 2000
Harness, Dickey & Pierce, P.L.C.
P.O. Box 828
Bloomfield Hills, Michigan 48303
(248) 641-1600

Gregory Stobbs
Gregory A. Stobbs
Reg. No. 28764

05 - 26 - 00

A

JC808 U.S. PTO
05/25/00

JC714 U.S. PTO
09/578739
05/25/00

HARNESS, DICKEY & PIERCE, P.L.C.
ATTORNEYS AND COUNSELORS
P.O. BOX 828
BLOOMFIELD HILLS, MICHIGAN 48303
U.S.A.

TELEPHONE
(248) 641-1600

TELEFACSIMILE
(248) 641-0270

Date: May 25, 2000

Hon. Commissioner of Patents and Trademarks
Washington, D.C. 20231

Sir:

EXPRESS MAILING CERTIFICATE

Applicant:	Ibrahim Mostafa Kamel Sarit Mukherjee Zongming Fei
Serial No. (if any):	__________ which is a continuation of patent application PCT International Serial No. 99/25422 filed 10/28/99 which priority was based on U.S. Provisional Application No. 60/106,167, filed 10/29/98.
For:	Providing VCR Functionality for Data-Centered Video Multicast
Docket:	9432-000057
Attorney:	Gregory A. Stobbs

"Express Mail" Mailing Label Number **EJ 486 600 570 US**

Date of Deposit **May 25, 2000**

I hereby certify and verify that the accompanying **check in the amount of $1,044.00 for filing fee; 3-page transmittal letter (in triplicate); 48-page Patent Application; copy of Declaration and Power of Attorney; five (5) sheets of drawings showing Figures 1-7 and return postcard** are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. § 1.10 on the date indicated above and are addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

Pamela Strauss

Pamela Strauss

PROVIDING VCR FUNCTIONALITY FOR DATA-CENTERED VIDEO MULTICAST

BACKGROUND AND SUMMARY OF THE INVENTION

The present invention relates generally to video on-demand multicasting. More particularly, the invention relates to a technique for providing VCR functionality (e.g., play, pause, fast forward, rewind, jump) in a data-centered video multicast or broadcast network.

Video on-demand service promises to create an entire new dimension in home entertainment possibilities. Whereas current pay-per-view broadcasts are confined to specific broadcast times, video on-demand relaxes adherence to strict schedules. Customers are able to request a specific video, selected from a database of choices, and have the video delivered essentially instantaneously or, in the worse case, after only a brief delay.

It is recognized that most requests for videos are from a small group of the currently most popular movies. Using multicast to send these movies is an efficient way to satisfy customer demand. Currently there are two basic approaches to providing multicast video on-demand. In the user-centered approach, the server allocates channels to one user or to a group of users and then sends the movie over the allocated channel or channels. In early unicast systems (a special case of multicast, with one user per group) a single channel was allocated

for each user. More recent multicast systems allocate one channel to support several users.

In the data-centered approach, the server allocates channels to a movie or to a part of a movie. Through periodic broadcasts, a given channel broadcasts a movie repeatedly in cycles. Conventionally there are several different schemes for broadcasting in the data-centered paradigm. Pyramid broadcasting divides a movie into segments of exponentially increasing size and lets each channel broadcast each segment repeatedly. The segment is broadcast in the channel at a faster speed than playback speed. Skyscraper broadcasting modifies the pyramid approach, using a different distribution of segment sizes and broadcasting at the same speed as playback speed. Skyscraper broadcasting places an upper bound on the maximum weight of the segment size. This is done to reduce storage requirements at the client side (user). The number of multicast channels required does not depend on the number of requests from users. Thus the data-centered approach appears to be more scalable than the user-centered approach.

The multicast approach to supplying video on-demand from several users at one time, to some extent, sacrifices special requirements of each individual user. The multicast systems usually require the client to wait for a certain period before it can be served. This is referred to as the *startup latency* for the services. VCR functions

such as pause, fast forward, fast rewind and jump to new location are difficult to provide in data-centered multicast systems. Although there have been attempts to provide VCR functionality, the solutions offered to date have not been able to provide scalability while at the same time guaranteeing seamless delivery. Typically, as the size of the system is scaled up, it becomes increasingly more difficult to provide seamless operation. Thus users experience jumpiness or breaks in delivery that detract from the enjoyment of the system. The present invention proposes a new scheme which can provide VCR functionality in data-centered multicast video on-demand systems. The scheme is scalable and is far more able to guarantee smooth, discontinuity-free delivery. The system employs separate buffers at each client, from which media content may be played back to support VCR functionality. A loader associated with each buffer downloads segments of the video stream from the video data server under control of a pre-fetch manager that intelligently determines what to pre-fetch into the buffer. The pre-fetch manager does this by monitoring the playback position pointer and controls the pre-fetching operation to maintain the playback position pointer within a predetermined range within the buffer.

The system also employs a calculation of feasible points based on a set of predefined rules. These feasible points are used in determining which operations are performed in a seamless fashion and also to identify when seamless operation may not be possible. In the

latter case, appropriate user-friendly performance is initiated so that the user does not experience undue jerkiness or long delays or gaps in media delivery.

For a more complete understanding of the invention, its objects and advantages, refer to the following specification and to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure **1** is block diagram of the presently preferred embodiment for providing VCR functionality;

Figure **2** illustrates a first broadcasting scenario, demonstrating how discontinuity occurs in the case of an unrestricted jump;

Figure **3** illustrates a similar broadcasting scenario, showing how discontinuity can occur in the case of a fastforward action;

Figure **4** illustrates Case 1 of the presently preferred set of rules for assessing feasible points;

Figure **5** illustrates Case 2 of the presently preferred rules for determining feasible points;

Figure **6** illustrates Case 3 of the presently preferred rules for assessing feasible points;

Figure **7** is an example showing how the feasible point analysis is performed in accordance with the presently preferred embodiment.

DESCRIPTION OF THE PREFERRED EMBODIMENT

A presently preferred embodiment of the system is illustrated in Figure **1**. The system includes a player module **10** that provides an associated user interface **12** by which the user can select among a set of VCR functions. Illustrated in Figure **1** are the following functions: jump backward **14**, fast rewind **16**, pause **18**, play **20**, slow motion play **22**, fast forward **24** and jump forward **26**. These VCR functions are exemplary of the type of functions that may be provided in a consumer product. Of course, other functions may also be provided, if desired.

For instance, a jog shuttle mechanism may be provided to allow the user to move forward or backward through successive frames of the video content. Also note that a stop function may also be included. However, in the presently preferred implementation, the pause function **18** provides nearly the same functionality as a stop function would. This is so because, unlike in a mechanical VCR, there is no requirement to unload the tape from the tape head or to halt the tape drive motors and capstan. Nevertheless, if desired, both pause and stop functions may be included, in which case the pause function might provide a freeze-frame function while the stop function would blank the screen, for example.

Player **10** accepts VCR commands from the user interface and plays back the video stream through a suitable monitor **30**. Player **10** obtains the video stream from one or more buffers **32**. In the presently

preferred embodiment there is a loader module **34** associated with each buffer. The loader module is responsible for interfacing with the distribution network **36** to download the required video content, as will be more fully described below.

The loader modules are controlled by a manager module **38** that is responsible for implementing the selective downloading strategy of the system. As will be more fully described below, the manager is able to access the playback pointer **40** of player **10**. This pointer shows the current playback location within the buffer **32** that is currently supplying the video stream to the player for display on the monitor **30**. The manager module **38** includes a feasible point calculation module or calculation processor **42** that is responsible for assessing which user-requested functions may result in discontinuous service. The feasible point calculation module thus supplies manager **38** with the information needed to interpret the user's request in a modified fashion if the original request would result in undesirable playback (e.g., jumpy or discontinuous playback or long delay).

When the client wishes to download video segments from a broadcast channel on network **36**, the manager **38** of the preferred embodiment causes loaders **34** to load information into the buffers in two phases. The first phase is a pyramid phase (unequal segments) during which all loaders are used to download information to guarantee smooth playback in the normal forward direction. Once the buffers **32**

have been loaded to support the initial forward play mode, manager **38** gradually phases the loaders **34** into the equal segment phase. In the equal segment phase, some of the segments may be fetched to support playback of backward segments (i.e., content that precedes the current playback pointer).

The manager selects the channels and segments to be loaded based on the current playback segment, as designated by the playback pointer **40**. The manager **38** assigns priority to each segment based on its proximity to the current segment and also based on whether loading is in the initial pyramid phase or in the later equal segment phase.

To illustrate, assume the current segment is k. If k is one of the pyramid phase segments, the first three segments in the forward direction (k, k+1, k+2) are assigned the highest priority. If k is one of the equal phase segments, the highest priority segments are the current segment k and those that immediately precede and follow it (k, k+1, k-1). The priorities of the remaining segments are determined by their distance from the current segment. The nearer to the current segment the higher its priority. In case of ties, the tie is broken in favor of the segment in the forward direction. If there are more than three loaders and buffers, allocation is based on the priority of the segment.

We will further consider here the special case where there are only three loaders and buffers. In this case manager **38** allocates or

reallocates the loaders into situations: (a) startup and (b) after performing a VCR action.

In the startup situation the playback pointer lies at the boundary of the first segment to be played. Note that the boundaries of all segments are also boundaries of the first segment at startup. At this boundary both de-allocation and allocation can occur. If a loader finishes loading a segment, it is returned to the free loader pool (de-allocation). This de-allocation process is voluntary by the loader. Allocation can also occur after de-allocation.

During the pyramid phase, the three loaders are allocated to the next unassigned segments in line. During the equal segment phase, if the play point is in the earlier half of the current segment (the current segment being designated segment k), the three loaders are assigned to segments k-1, k, k+1. If the play point is in the latter half of the current segment, the three loaders are assigned to segments k, k+1, k+2. If the contents of the segment to be played is already in the buffer no loading occurs for that segment. Later downloading segments may overwrite earlier segments already in the buffer. However, if the playpoint is in to-be-overwritten segment in the buffer, the manager **38** prevents it from being overwritten beyond the playpoint. We call this playpoint protection.

In the second situation (after every VCR action is performed) a somewhat different sequence proceeds. During the pyramid phase, the

manager checks whether the loaders are assigned to k, k+1, k+2. If this is the case, no actions are taken. Otherwise, the manager will enforce preemptively the assignment. During the equal segment phase, if the playpoint is inside the buffer, no actions are taken. The relative position of the playpoint is adjusted when the downloading point (channel broadcast point) comes to the boundaries of the equal segments. If the playpoint goes outside the buffer (the current segment assumed to be k), the manager reallocates the loaders to segments k, k+1 and k-1. This may cause an involuntary de-allocation of loaders.

The manager **38** of the preferred embodiment controls the loading of buffers **32**, based on the playback pointer **40**, striving to maintain the playback pointer within a pre-determined range within the buffer. In the presently preferred embodiment, the manager causes information to be fetched and stored in the buffers so that the playback pointer remains at or near the center of the buffer after the initial pyramid phase loading is completed. Doing this allows the system to provide continuous VCR functionality with a very high probability that the user's request can be smoothly carried out. In contrast with other schemes, the present system empowers the client to actively pre-fetch nearby frames around the playback point and to locally manufacture the contents required to implement most VCR actions by appropriately accessing the local buffers **32**.

To account for situations in which a user's VCR command may not be feasible, manager **38** includes a feasible point calculation module **42**. The preferred implementation employs a set of rules to decide whether a given destination point is feasible or not, and how to respond accordingly. To provide high quality VCR functionality and seamless playback, the system strives to guarantee that the user can jump to a desired destination point p and then resume normal playback to the end of the movie without discontinuity. Given the particular channel broadcasting situation and the buffer conditions at the client, not all points are feasible points. For example, as illustrated in Figure **2**, an unrestricted jump can lead to discontinuities of later playback even if there is no subsequent VCR action. As Figure **2** illustrates, assume that a user issues a VCR action to jump to position c of segment k outside of the client's buffer and that the channel k happens to broadcast at point c. If the jump is allowed, after an elapse of time t, the buffer will only have c_k e_k and c_{k+1} d, however the client needs to playback from e_k (or b_{k+1}) to c_{k+1}, which is not available and results in discontinuous playback.

Even if the video frame contents are in the buffer, some VCR actions can lead to discontinuity after normal play is resumed. Figure **3** illustrates this. In Figure **3** a fastforward action is initiated. Assume that the channel broadcasting point is c_i for channel i. The contents of channel k are in the buffer and contents of channel k+1 are not in the

buffer. The current playback point is b_k. If a fastforward command is issued with length of e_k e_k, all the contents are in the buffer. Assume that the fastforward of this length takes time t. If the system performs this action to the full length, it will take time t to the playback position of e_k. At this point, only b_k e_k and c_{k+1} d in the buffer, but b_{k+1} c_{k+1} is needed to continue normal play and is not available.

To address the issues illustrated in Figures **2** and **3**, the feasible point calculation module determines whether a given point is feasible or not and then determines how to find the nearest point that is feasible. The presently preferred embodiment distinguishes three following cases according to the relative size of the segments involved.

Assume that the size of segment i is s_i. The beginning position and end position of segment i and b_i and e_i, respectively. We assume that $b_i = 0$ for all $1 \leq i \leq K$. At any time each channel is broadcasting a specific frame in the segment. This point is denoted as c_i and we have $b_i \leq c_i \leq e_i$.

If the destination point p is located at or before the channel point c_i, we call the segment containing the desired destination current segment and we assume it is segment k and broadcast in channel k. First we give the rules for this case. After that, we consider the cased in which the destination point p is located after the channel point c_i.

- Case 1 (Figure **4**): $s_k = s_{k+1}$. This is the case in which the size of the current segment is equal to the size of the next segment. It is abbreviated as (x, x) case.

  - If pc_k is in the buffer, then p is feasible; otherwise, the later nearest feasible point is the point q such that qc_k are in the buffer with the smallest q value.

- Case 2 (Figure **5**): $s_{k+1} = 2 * s_k$ and $s_{k+2} = s_{k+1}$. This is the case in which the size of the current segment is half of the size of the next segment and the next two segments are of equal size. It is abbreviated as (x, 2x, 2x) case.

  - case 2.1 $c_k = c_{k+1}$. This is case in which the broadcasting position of channel k is the same as channel k + 1, or they are left aligned.

    If pc_k and $b_{k+1}c_{k+1}$ are in the buffer, then p is feasible: otherwise, if $b_{k+1}c_{k+1}$ is in the buffer, the later nearest feasible point is the point q such that qc_k are in the buffer with the smallest q value; if $b_{k+1}c_{k+1}$ is not in the buffer, the later nearest feasible point is the point q such that qc_{k+1} are in the buffer with the smallest q value;

  - case 2.2 $c_k \neq c_{k+1}$. This is case in which the broadcasting position of channel k is not the same as channel k + 1, or they are left aligned.

If pc_k is in the buffer, then p is feasible; otherwise, the later nearest feasible point is the point q such that qc_k are in the buffer with the smallest q value.

- Case 3 (Figure **6**): $s_{k+1} = 2 * s_{k+1}$ and $s_{k+2} = 2 * s_{k+1}$. This is the case in which the size of the current segment is half of the next segment and the size of next segment is in turn half of the size of its next segment. It is abbreviated as (x, 2x, 4x) case.

  - case 3.1 $c_k = c_{k+1} = c_{k+2}$. This is case in which the broadcasting position of channels k, k + 1 and k + 2 are the same, or they are left aligned.

If pc_k, $b_{k+1}c_{k+1}$ and $b_{k+2}c_{k+2}$ are in the buffer, then p is feasible; otherwise, if $b_{k+1}c_{k+1}$ and $b_{k+2}c_{k+2}$ are in the buffer, the later nearest feasible point is the point q such that qc_k are in the buffer with the smallest q value; if $b_{k+2}c_{k+2}$ is in the buffer and $b_{k+1}c_{k+1}$ is not all in the buffer, the later nearest feasible point is in the point q such that qc_{k+1} is in the buffer with the smallest q value: if $b_{k+1}c_{k+1}$ is not in the buffer, the later nearest feasible point is the point q such that qc_{k+2} is in the buffer with the smallest q value:

  - case 3.2 $c_k = c_{k+1} \neq c_{k+2}$. This is case in which the broadcasting position of channels k, k + 1 are the same, but not the same as that of channel k + 2.

If pc_k, $b_{k+1}c_{k+1}$ are in the buffer, then p is feasible; otherwise, if $b_{k+1}c_{k+1}$ is in the buffer, the later nearest feasible point is

the point q such that qc_k is in the buffer with the smallest q value; if $b_{k+1}c_{k+1}$ is not all in the buffer, the later nearest feasible point is the point q such that qc_{k+1} is in the buffer with the smallest q value: if $b_{k+1}c_{k+1}$ is not in the buffer, the later nearest feasible point is the point q such that qc_{k+1} is in the buffer with the smallest q value.

- case 3.3 $c_k \neq c_{k+1} \neq c_{k+2}$. This is case in which the broadcasting position of channels k + 1, k + 2 are the same, but not the same as that of channel k.

If pc_k, $b_{k+2}c_{k+2}$ are in the buffer, then p is feasible; otherwise, if $b_{k+2}c_{k+2}$ is in the buffer, the later nearest feasible point is the point q such that qc_k is in the buffer with the smallest q value; if $b_{k+2}c_{k+2}$ is not all in the buffer, the later nearest feasible point is the point q such that qc_{k+2} is in the buffer with the smallest q value.

- case 3.4 $c_k \neq c_{k+1} \neq c_{k+2}$. This is case in which the broadcasting position of channels k, k +1 and k + 2 are all different, or they are all right-aligned.

If pc_k is in the buffer, then p is feasible; otherwise, the later nearest feasible point is the point q such that qc_k are in the buffer with the smallest q value.

If the destination point p is located after the channel point c_i, the next segment to the one that contains the desired destination is called the current segment. We assume it is segment k. In this situation, pc_k

means pe_{k-1} and b_kc_k. After this modification, we can use the above rules.

Figure **7** shows an example how to determine the feasible point in this case. Assume the destination point is p, which is later than the broadcasting point c_7. So the current segment should be k = 8. Since the segment size of current segment is half next segment and next two segments are of equal size, this is the case 2.1 of the above rule. As stated above, we use pc_k to represent pe_7 and b_8c_8. According to the rule 2.1, p is a feasible point in case (a). In case (b), p is not a feasible point and the nearest feasible point is d because $b_{k+1}c_{k+1}$ is in the buffer and d is the smallest value such that dc_k is in the buffer.

The feasible point calculations, described above, allow the user to specify an arbitrary destination within the video-on-demand data stream. The system automatically adjusts the destination to a nearby feasible point, if the point requested by the user is not feasible. Thus, when the destination is not in the buffer, only those destination frames that are being broadcast at each channel are available immediately. The feasible point calculation module assesses what is possible and selects the nearest frame among those available as the adjusted destination if the requested destination cannot be met. If the feasible point calculation module determines that the requested frame is feasible, we call this an allowed destination. If the feasible point calculation module determined that the requested frame cannot be met

without discontinuity, it substitutes a different frame that can be met. We call this the adjusted destination.

One desirable property of the destination, either allowed or adjusted, is that once the client resumes normal play from this destination, the client is able to playback to the end of the video without experiencing interruptions, provided there are no further VCR actions.

In developing an actual embodiment of the invention, the system designer may wish to specify the number of buffers used at each client. To implement a robust VCR-oriented scheme, two conditions should be preferably met:

(1) If the destination segment is not in the buffer, and we still require that we can always select the frame being broadcast at this channel as the destination, the segments sized must be equal to the size of the next segment in order to guarantee that the client can playback the next segment when it finishes playing the current segment.

(2) The sizes of all segments beginning from the latter of these two equal segments must satisfy the continuity condition that guarantees smooth playback of the remainder of the video will be possible, assuming the client can download from m channels simultaneously.

Equation 1 below presents the function f(n) that will satisfy the above two conditions. The equation assumes that a client can download from m channels simultaneously.

<u>Equation 1</u>

$$f(n) = 2^{n - \left[\frac{n}{m}\right]}$$

As an example, we give below the series generated by f(n) with m = 5:

1, 2, 4, 8, 16, 16, 32, 64, 128, 256, 256, 512, 1024, 2048, 4096,...

In designing a system, the value of m (simultaneously available channels) places a minimum requirement on the bandwidth needed at each client. Considering the heterogeneity of the client, we like m to be small. On the other hand, in order to pre-fetch nearby segments of the current playback segment, clients download both the next segment for forward VCR action and the previous segment for backward VCR action. These segments are fetched in addition to the current segment. Thus in the presently preferred embodiment the value of m is preferably 3 or greater. If fewer than 3 channels are available simultaneously, then the system will have to sacrifice in supporting forward or backward VCR functionality. Thus in the preceding example, we have used

m = 3 to illustrate both forward and backward VCR functionality. Of course, in a practical implementation, the value selected for m can be greater than 3 while still preserving the benefit of supporting both forward and backward action.

Given that the series dictated by Equation 1 is monotonically increasing and is bounded by f(u) = f(K), we determine the buffer requirement for a client to be the size of the largest segment minus one size unit. Thus the buffer requirement may be expressed as a function of f(n) as follows:

Equation 2

$$Buffer = \frac{f(u)xL}{\sum_{n=1}^{K} f(n)}.$$

In operation, when a client downloads video segments from the broadcast channels, we shall assume that at the outset no VCR functions are invoked and the video is to proceed at the beginning of the program. At the beginning, three loaders wait until the boundary of the first segment is detected. This waiting time is the startup latency and the maximum waiting time will be the time required to playback the first segment. The first loader begins to download from the first channel and contents of its buffer are played. The other two loaders are assigned to the second and third channels, respectively. Depending on the relative positions of each channel, the loader may download from the second channel at the beginning or at the end of the first segment.

Similarly, the loader may download from the third channel at the beginning or end of the second segment. When the contents of the buffers are played they are immediately discarded. Whenever a loader finishes downloading the assigned channel, it is assigned to the next immediate unassigned channel. It begins downloading from the channel when the length of the contents in its buffer is less than the assigned channel segment size. This guarantees that downloading of a segment is delayed as much as possible to save buffer space, but it also guarantees smooth playback of that segment.

The foregoing system is thus able to provide smooth, continuous playback while supporting all popular VCR functions. While the invention has been described in its presently preferred embodiments, it will be understood that the invention is capable of modification without departing from the spirit of the invention as set forth in the appended claims. For example, the number of buffers employed at a client location can vary. Also, the number of loaders used to service the buffer population can also vary. While a one-to-one correspondence between buffers and loaders is presently preferred, some implementations may be able to provide adequate throughput with fewer loaders than buffers.

CLAIMS

What is claimed is:

1. A system for supporting at least one VCR function in a network-based video-on-demand delivery system, comprising:

a player having a user interface that provides at least one user-actuable VCR function initiator, said player being adapted for coupling to a display monitor to supply a video stream to said monitor for playback;

said player maintaining at least one playback pointer that provides information indicative of the current video playback frame;

at least one buffer coupled to said player having an associated loader for downloading video data from said delivery system;

a manager coupled to said player and to said loader for selectively causing said loader to download video data from said delivery system in order to maintain said playback pointer within a predetermined location range within said buffer.

2. The system of claim **1** further comprising a plurality of buffers each being selectively loaded with video data under control of said manager such that at least a one of said buffers contains video data that precedes the current video playback frame.

3. The system of claim **2** wherein each of said buffers has an associated loader responsible for supplying that buffer with downloaded video data.

4. The system of claim **1** wherein said manager includes a feasible point calculation module that assesses whether the destination point resulting from a selected user interaction with said VCR function initiator will result in discontinuous playback.

5. The system of claim **4** wherein said manager is responsive to said feasible point calculation module to modify a requested VCR function such that said function will not result in discontinuous playback.

6. The system of claim **1** wherein said user interface provides VCR functions selected from the group consisting of: jump backward, fast rewind, pause, stop, play, slow motion play, fast forward and jump forward.

7. The system of claim **1** further comprising at least three buffers coupled to said player.

8. The system of claim **1** further comprising at least three buffers coupled to said player, each buffer having an associated loader.

9. The system of claim **1** wherein said manager implements at least two different downloading schemes, including a first scheme for loading said buffer upon startup and a second scheme for loading said buffers after startup.

10. The system of claim **1** further comprising at least three buffers coupled to said player and wherein said manager implements at least two different downloading schemes, including:

(a) a first downloading scheme in which a first one of said buffers is loaded with a first segment and the second and third of said buffers are respectively loaded with second and third segments that each follow the first segment;

(b) a second downloading scheme in which a first one of said buffers is loaded with a first segment, a second of said buffers is loaded with a second segment that precedes the first segment, and a third of said buffers is loaded with a third segment that follows said first segment.

11. A Video-on Demand client system to prefetch segments of video data streams through multiple communications channels of data-centered broadcasting network from a video data server for implementing VCR functions including at least playback the segments, said client system comprising:

at least one loader to download the segments of the video stream from the video data server;

at least one buffer to store the downloaded segments from said loader;

a player to playback the segments read from said buffer, said player being responsive to VCR function commands given through user-interface thereof;

a playback pointer to issue playback commands to said player for designating a playback starting point of the segment in said buffer; and

a prefetch manager to issue prefetch commands to said loader for prefetching the segments from the server based on the current playback point of the segment in said buffer so as to keep the playback point designated by said pointer within predetermined range of said buffer.

12. The client system as in claim **11**, wherein the predetermined range of said buffer is a middle part thereof.

13. The client system as in claim **11**, wherein each number of said loaders and said buffers are respectively at least three.

14. The client system as in claim **11**, the VCR function given through the user-interface of said player including normal play which playbacks the segments of the video streams at normal speed, fast forward which playbacks the segments at multiple times speed as normal play in forward direction, fast backward which playbacks the segments at multiple times speed as the normal play in backward direction, slow forward which playbacks the segment slower than the normal play in forward direction, pause which playbacks stationary with keeping the current playback point, jump forward which jumps directory to the destination point of the segment specified in terms of forward distance relative to the current playback point and resumes the normal play from the jumped point, and jump backward which jumps directory to the destination point of the segment specified in terms of backward distance relative to the current playback point and resumes the normal play from the jumped point.

15. The client system as in claim **11**, wherein according to size of the broadcasting segment through the channels of the network from the server to the client, said playback pointer designates the feasible playback starting point for a destination frame point p of the segment designated by VCR function commands.

16. The client system as in claim **15**, where

- k is defined as natural number;
- b(k) is defined as a beginning frame point of segment No. k;
- e(k) is defined as an end frame point of segment No. k;
- c(k) is defined as a current broadcasting frame point of segment No. k;
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;
- point value of b(k) equals to 0, (1 <= k <= K);
- condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k; and
- condition (b): the size of the current broadcasting segment k equals to the size of next segment k+1,

wherein

when the conditions (a) and (b) are satisfied, if frames between the destination point p and c(k) is in the buffer, then the point p is the feasible point, otherwise the later nearest point q that frames between points of q and c(k) are in the buffer is the feasible point having smallest point value.

17. The client system as in claim **15**, where

- k is defined as natural number;
- b(k) is defined as a beginning frame point of segment No. k;
- e(k) is defined as an end frame point of segment No. k;
- c(k) is defined as a current broadcasting frame point of segment No. k;
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;
- point value of b(k) equals to 0 as an offset value, (1 <= k <= K);
- condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;
- condition (c): the size of the current broadcasting segment k is half of size of the next segment k+1 and the size of the next two segments k+1 and k+2; and
- condition (d): the point value of c(k) equals to the point value of c(k+1),

wherein

when the conditions (a), (c) and (d) are satisfied,

if both frames between the destination point p and

c(k) and frames between the point b(k+1) and c(k+1) are in the buffer, then the point p is the feasible point,

otherwise:

if frames between the points of b(k+1) and c(k+1) are in the buffer, then the later nearest point q that frames between q and c(k) is in the buffer is the feasible point having smallest point value,

if frames between the points of b(k+1) and c(k+1) are not in the buffer, then the later nearest point q that frames between the points q and c(k+1) is in the buffer are the feasible point having smallest point value.

18. The client system as in claim **15**, where

- k is defined as natural number;
- b(k) is defined as a beginning frame point of segment No. k;
- e(k) is defined as an end frame point of segment No. k;
- c(k) is defined as a current broadcasting frame point of segment No. k;
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;
- point value of b(k) equals to 0 as an offset value, (1 <= k <= K);
- condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;
- condition (c): the size of the current broadcasting segment k is half of size of the next segment k+1 and the size of the next two segments k+1 and k+2; and
- condition (e): the point value of c(k) does not equals to the point value of c(k+1),

wherein

when the conditions (a), (c) and (e) are satisfied,

if both frames between the destination point p and

c(k) are in the buffer, then the point p is the feasible point, otherwise the later nearest point q that frames between points of q and c(k) are in the buffer is the feasible point having smallest point value.

19. The client system as in claim **15**, where

- k is defined as natural number;
- b(k) is defined as a beginning frame point of segment No. k;
- e(k) is defined as an end frame point of segment No. k;
- c(k) is defined as a current broadcasting frame point of segment No. k;
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;
- point value of b(k) equals to 0 as an offset value, (1 <= k <= K);
- condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;
- condition (f): the size of the current broadcasting segment k is half of size of the next segment k+1 and the size of the next segments k+1 is half of size of its next segment k+2; and
- condition (g): the point value of c(k) equals to the both point values of c(k+1) and c(k+2),

wherein

when the conditions (a), (f) and (g) are satisfied,

if frames between the destination point p and c(k), frames

between the point b(k+1) and c(k+1), and frames between the point b(k+2) and c(k+2) are in the buffer, then the point p is the feasible point,

otherwise:

if frames between the point b(k+1) and c(k+1), and frames between the point b(k+2) and c(k+2)are in the buffer, then the later nearest point q that frames between q and c(k) is in the buffer is the feasible point having smallest point value,

if frames between the points of b(k+1) and c(k+1) are not all in the buffer, and frames between the point b(k+2) and c(k+2)are in the buffer, then the later nearest point q that frames between q and c(k+1) is in the buffer is the feasible point having smallest point value,

if frames between the points of b(k+1) and c(k+1) are not in the buffer, then the later nearest point q that frames between the points q and c(k+2) are in the buffer is the feasible point having smallest point value.

20. The client system as in claim **15**, where

• k is defined as natural number;

• b(k) is defined as a beginning frame point of segment No. k;

• e(k) is defined as an end frame point of segment No. k;

• c(k) is defined as a current broadcasting frame point of segment No. k;

• K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;

• point value of b(k) equals to 0 as an offset value, (1 <= k <= K);

• condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;

• condition (f): the size of the current broadcasting segment **k** is half of size of the next segment k**+1** and the size of the next segments k**+1** is half of size of its next segment k+2; and

• condition (h): the point value of c(k) equals to the both point values of c(k+1) and does not equal to c(k+2),

wherein

when the conditions (a), (f) and (h) are satisfied,

if frames between the destination point p and c(k),

and frames between the point b(k+1) and c(k+1) are in the buffer, then the point p is the feasible point, otherwise:

if frames between the point b(k+1) and c(k+1) are in the buffer, then the later nearest point q that frames between q and c(k) is in the buffer is the feasible point having smallest point value,

if frames between the points of b(k+1) and c(k+1) are not all in the buffer, then the later nearest point q that frames between q and c(k+1) is in the buffer is the feasible point having smallest point value.

21. The client system as in claim 15, where

• k is defined as natural number;

• b(k) is defined as a beginning frame point of segment No. k;

• e(k) is defined as an end frame point of segment No. k;

• c(k) is defined as a current broadcasting frame point of segment No. k;

• K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;

• point value of b(k) equals to 0 as an offset value, (1 <= k <= K);

• condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;

• condition (f): the size of the current broadcasting segment k is half of size of the next segment k+1 and the size of the next segments k+1 is half of size of its next segment k+2; and

• condition (i): the point value of c(k) equals to the point values of c(k+2) and does not equal to c(k+1),

wherein

when the conditions (a), (f) and (i) are satisfied,

if frames between the destination point p and c(k), and

frames between the point b(k+2) and c(k+2) are in the buffer, then the point p is the feasible point,

otherwise:

if frames between the point b(k+2) and c(k+2) are in the buffer, then the later nearest point q that frames between q and c(k) is in the buffer is the feasible point having smallest point value,

if frames between the points of b(k+2) and c(k+2) are not all in the buffer, then the later nearest point q that frames between q and c(k+2) is in the buffer is the feasible point having smallest point value.

22. The client system as in claim **15**, where

• k is defined as natural number;

• b(k) is defined as a beginning frame point of segment No. k;

• e(k) is defined as an end frame point of segment No. k;

• c(k) is defined as a current broadcasting frame point of segment No. k;

• K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;

• point value of b(k) equals to 0 as an offset value, (1 <= k <= K);

• condition (a): the destination point p is located at or before broadcasting point c(k) of the segment k broadcasting in channel k;

• condition (f): the size of the current broadcasting segment k is half of size of the next segment k+1 and the size of the next segments k+1 is half of size of its next segment k+2; and

• condition (j): the point value of c(k) does not equal to the both point values of c(k+1) and c(k+2),

wherein

when the conditions (a), (f) and (j) are satisfied,

if frames between the destination point p and c(k) are in the buffer, then the point p is the feasible point,

otherwise the later nearest point q that frames between q and c(k) is in the buffer is the feasible point having smallest point value.

23. The client system as in claim **15**, where

- k is defined as natural number;
- b(k) is defined as a beginning frame point of segment No. k;
- e(k) is defined as an end frame point of segment No. k;
- c(k) is defined as a current broadcasting frame point of segment No. k;
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length;
- point value of b(k) equals to 0 as an offset value, (1 <= k <= K);
- condition (k): the destination point p is located after the current broadcasting point c (k) of the segment k broadcasting in channel k, wherein

when the conditions (k) is satisfied, frames between the destination point p and c(k) are considered as frames between p and e(k-1) and frames between b(k) and c(k) so that the destination point p is considered to be located at or before broadcasting point c(k) of the segment k broadcasting in channel k.

24. A method of demanding segments of video data streams by a Video-on Demand client system through multiple communications channels of data-centered broadcasting network from a video data server for implementing VCR functions including at least playback the segments by a player of the client system, said method comprising the steps of:

(a) downloading the segments of the video stream from the video server to at least one loader;

(b) storing the downloaded segments of the loader at least one buffer for being read to playback by a player therefrom;

(c) issuing at least playback command to the player for designating a playback starting point of the segment in the buffer; and

(d) issuing prefetch commands to the loader for prefetching the segments from the server based on the current playback point of the segment so as to keep the playback point designated by said pointer within predetermined range of a buffer.

25. A method for supplying segments of video data streams to Video-on-Demand clients through multiple communications channels of data-centered broadcasting network from a video data server in a Video-on-Demand system, said method comprising:

establishing the communications channel between the client and the server;

dividing each video data stream into multiple sequential segments in accordance with equation: SIZE = {f(N)*L}/{sigma function of f(J), J=1~K};

where:

- SIZE is defined as size of segment No. N;
- N is defined as natural number;
- f(N) is defined as 2 to the power {N – (N/M)};
- M is defined as a number of channels being able to download the segments simultaneously;
- L is defined as length of a set of data streams of each video;
- J is defined as natural number; and
- K is defined as numbers of channels and divided into segments of a set of video data streams of each video of length L,

and

transmitting the divided segments of each video multiple times to the clients through the multiple channels from the server.

26. The method of claim **25** further comprising the steps of :

receiving at the Video-on-Demand client the transmitted segments from the server;

playing the segments at the Video-on-Demand client; and

concurrently with playing and receiving, storing the transmitted sequential segments at the Video-on-Demand client.

27. A Video-on Demand broadcasting system, said system comprising:

a video data server to supply video data streams to multiple communications channels of data-centered broadcasting network;

Video-on-Demand clients through to receive segments of the video data streams through the network

wherein after establishing the communications channel between the client and the server,

said server having dividing means for dividing each video data stream into multiple sequential segments in accordance with equation:

SIZE = {f(N)*L}/{sigma function of f(J), J=1~K};

where:

- SIZE is defined as size of segment No. N;
- N is defined as natural number;
- f(N) is defined as 2 to the power {N – (N/M)};
- M is defined as a number of channels being able to download the segments simultaneously;
- L is defined as length of a set of data streams of each video;

• J is defined as natural number; and

• K is defined as numbers of channels and

divided into segments of a set of video data streams of each video of length L,

and

transmitting means for transmitting the divided segments of each video multiple times to the clients through the multiple channels.

28. The system of claim **27** wherein the client having receiving means for receiving the transmitted segments from the server and playing means for playing the segments, and concurrently with playing and receiving, storing means for storing the transmitted sequential segments.

29. A video data server of the system as in claim **27**.

30. A Video-on-Demand client of the system as in claims **27**.

31. A Video-on-Demand client of the system as in claims **28**.

PROVIDING VCR FUNCTIONALITY FOR DATA-CENTERED VIDEO MULTICAST

ABSTRACT OF THE DISCLOSURE

The client application provides VCR functionality in a data-centered video multicast network (36) through active prefetching of the fragments from broadcast channels. The player (10) has a user interface (12) that supports a plurality of VCR functions (14-26). Plural buffers (32) and their associated loaders (34) download video data under control of a manager module (38). The manager module accesses the playback pointer (40) and employs feasible point calculation module (42) to cause the loaders to selectively download segments so that VCR functions can be supported while maintaining continuous playback after the VCR function is performed.

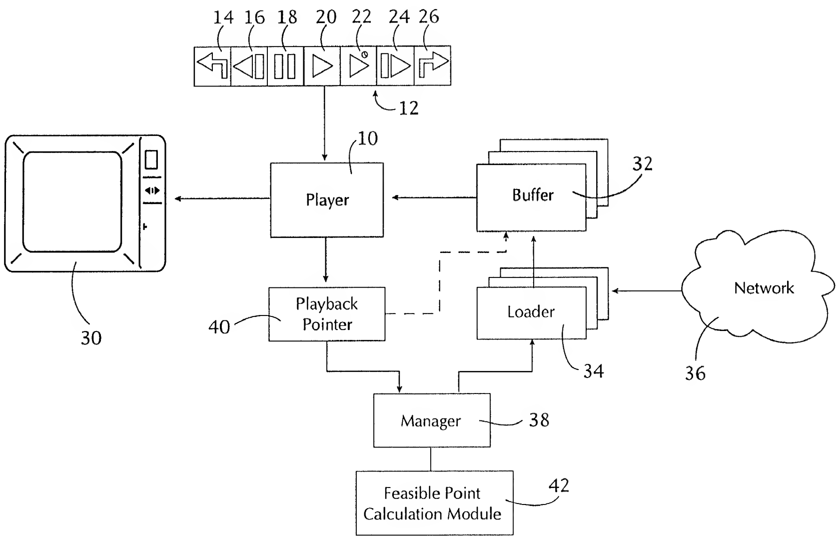
14
16
18
20
22
24
26
12
10
Player
Buffer
32
Playback
Pointer
40
30
Loader
34
Network
36
Manager
38
Feasible Point
Calculation Module
42

FIG. 1

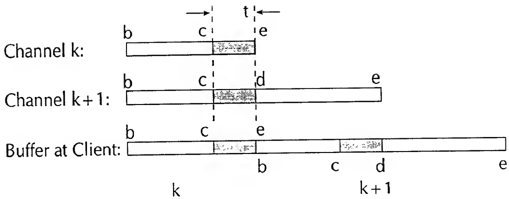
t
b
c
e
Channel k:
b
c
d
e
Channel k+1:
b
c
e
Buffer at Client:
b
c
d
e
k
k+1

FIG. 2

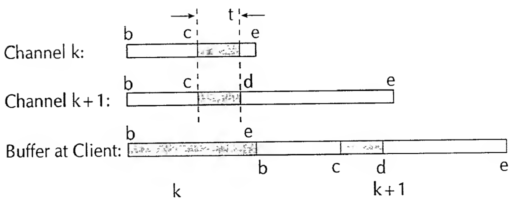
t
b
c
e
Channel k:
b
c
d
e
Channel k+1:
b
e
Buffer at Client:
b
c
d
e
k
k+1

FIG. 3

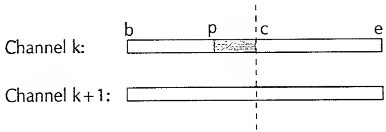
b
p
c
e
Channel k:
Channel k+1:

FIG. 4

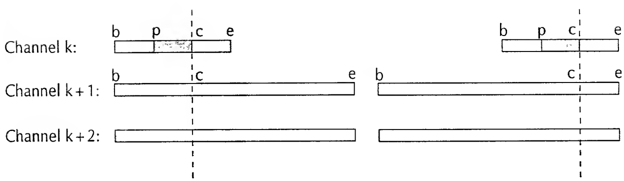
Channel k:
Channel k+1:
Channel k+2:
b
p
c
e

FIG. 5

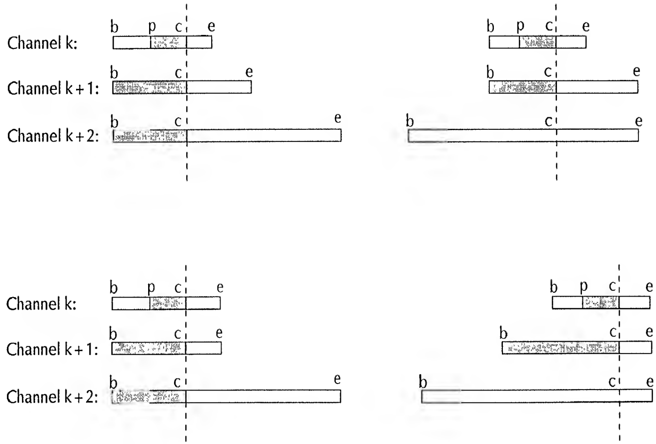
Channel k:
Channel k+1:
Channel k+2:
b
p
c
e
Channel k:
Channel k+1:
Channel k+2:

FIG. 6

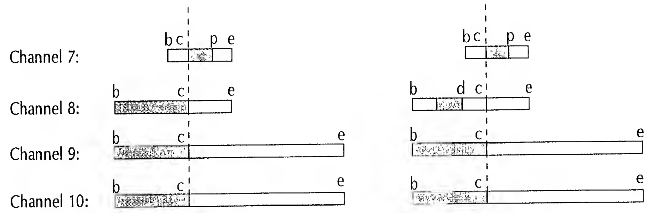

FIG. 7

DECLARATION AND POWER OF ATTORNEY

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am an original and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled

On Providing VCR Functionality for Data-Centered Video Multicast

the specification of which (check one)

[X] is attached hereto.

[ ] was filed on ______________________________ as Application Serial No. ____________________ and was amended on ______________________________________ (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application or to the patentability of the invention claimed therein in accordance with Title 37, Code of Federal Regulations, section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, section 119(a)-(d) of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

PRIOR FOREIGN APPLICATION(S)

			Priority Claim	
(Number)	(Country)	(Day/Month/Year filed)	Yes	No
(Number)	(Country)	(Day/Month/Year filed)	Yes	No
(Number)	(Country)	(Day/Month/Year filed)	Yes	No

DECLARATION AND POWER OF ATTORNEY

I hereby claim the benefit under Title 35, United States Code, §119(e) of any United States Provisional application(s) listed below:

PRIOR PROVISIONAL APPLICATIONS

(application serial number) (Month / Day / Year filed)

(application serial number) (Month / Day / Year filed)

I hereby claim the benefit under Title 35, United States Code, section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

Application Serial No.	Filing Date	Status – patented, pending, abandoned

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

I hereby appoint Gregory A. Stobbs, Reg. No. 28764, and each principal, attorney of counsel, associate and employee of Harness, Dickey & Pierce, P.L.C., who is a registered Patent Attorney, my attorney with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith. I request the Patent and Trademark Office to direct all correspondence and telephone calls relative to this application to Harness, Dickey & Pierce, P.L.C., P. O. Box 828, Bloomfield Hills, Michigan 48303 (248) 641-1600.

Full name of first inventor: Ibrahim Kamel

Inventor's signature: 10/28/98

Date: 10/28/98

Residence: 4131 Bayberry Court, Monmouth Junction, New Jersey 08852

Citizenship: EGYPT

Post Office Address: 4131 Bayberry Court, Monmouth Junctin, New Jersey 08852

DECLARATION AND POWER OF ATTORNEY

Full name of second joint inventor: Sarit Mukherjee

Second Inventor's signature:

Date: 10/28/98

Residence: 10423 Maple Leaf Drive, Lawrenceville, New Jersey 08648

Citizenship: INDIA

Post Office Address: 10423 Maple Leaf Drive, Lawrenceville, New Jersey 08648

Full name of third joint inventor, if any:

Third Inventor's signature:

Date:

Residence:

Citizenship:

Post Office Address:

Full name of fourth joint inventor, if any:

Fourth Inventor's signature:

Date:

Residence:

Citizenship:

Post Office Address:

Full name of fifth joint inventor, if any:

Fifth Inventor's signature:

Date:

Residence:

Citizenship:

Post Office Address: